Practical Set – 3

Program 3.1: Write python program to check whether the given list is palindrome or not using 1. reverse() method 2. Slicing.

Code:

```
list1 = [1, 2, 3, 4, 1]
list2 = list1
list1=list1.reverse()
if list2 == list1:
   print("Given list is palindrome..")
else:
   print("Given list is not palindrome..")
```

Output:

```
Given list is not palindrome..
```

Code:

```
list1 = [1, 2, 3, 2, 1]
if list1[::-1] == list1:
   print("Given list is palindrome..")
else:
   print("Given list is not palindrome..")
```

Output:

```
Given list is palindrome..
```

Program 3.2: Take a list [‘a’, ‘b’, ‘c’, ‘d’] & enumerate it using enumerate() function.

Code:

```
list1 = ['a', 'b', 'c', 'd']
l_type = enumerate(list1)
print("Type of list1 :", type(l_type))
print(list(enumerate(list1)))
print(list(enumerate(list1, 2)))
```

Output :

```
Type of list1 : <class 'enumerate'>
[(0, 'a'), (1, 'b'), (2, 'c'), (3, 'd')]
[(2, 'a'), (3, 'b'), (4, 'c'), (5, 'd')]
```

Program 3.3: Insert 10 numbers (taken from user) in a sorted list. [[Sorted list means a list which is always sorted: Don't user sort() method of list.]]

Code:

```
list1 = []
print("Enter 10 elements : ")
for i in range(10):
   value = int(input())
   for i in range(len(list1)):
      if list1[i] > value:
         break
   list1 = list1[:i] + [value] + list1[i:]

print(list1)
```

Output:

```
Enter 10 elements :
75
25
4
3
9
1
56
43
21
69
[1, 3, 4, 9, 21, 25, 43, 56, 69, 75]
```

Program 3.4: Consider sentence: "Hi Hello How are you ?". Display a list of all word which starts with letter 'H'.

Code:

```
str = "Hii ! Hello ! How are you ?"
letter = 'H'
lst = str.split(" ")

result = []
for i in lst:
   if i[0] == letter:
      result.append(i)
print(result)
```

Output:

```
['Hii', 'Hello', 'How']
```

Program 3.5: Given a sentence, return a sentence with the words reversed using join() & split(). Ex. 'How are you' --> 'you are How'

Code:

```
sentence = "How are you"
split_sentence = sentence.split(" ")
reverse_sentence = " ".join(reversed(split_sentence))
print(reverse_sentence)
```

Output:

```
you are How
```

Program 3.6: Flatten a nested list structure.
Example: if list1 = [1, [2, 3], [4, 5, [6, 7]]] then try to convert it in 1-dimensional [1, 2, 3, 4, 5, 6, 7].

Code:

```
list1 = [1, [2, 3], [4, 5, [6, 7]]]
list2 = []

def flat_list(lst):
    for i in lst:
        if type(i) == list:
            flat_list(i)
        else:
            list2.append(i)

print("Original list : ", list1)
flat_list(list1)
print("Resulting list : ", list2)
```

Output:

```
Original list :  [1, [2, 3], [4, 5, [6, 7]]]
Resulting list :  [1, 2, 3, 4, 5, 6, 7]
```